Claims

What is claimed is:

1. Computer-automated electronic design methodology comprising the steps of: accessing a first file comprising an algorithm that is described in an array-oriented programming language such as MATLAB; andgenerating a second file comprising a digital circuit representation that is synthesized automatically from the algorithm, where the representation is Register Transfer level VHDL or Verilog.

2. The methodology of Claim 1 wherein:

the second file comprises a design feature for optimization that is described in an intermediate format.

3. The methodology of Claim 1 wherein:

the second file is generated using a user directive, effectively allowing a user to guide a compiler by specifying a variable type or shape for optimization.

4. In an electronic design system comprising a compiler for synthesizing a circuit definition from a high-level definition, a process comprising the step of:

determining by a compiler a type or a shape of a variable or an intermediate temporary variable.

5. Electronic design scalarization process comprising the step of:

transforming a first intermediate-format description into a second intermediate-format description by translating an array statement in the first intermediate-format description into a loop in the second intermediate-format description.

6. In an automated design system comprising a processor for optimizing synthesis, a precision-inferencing method comprising the step of:

determining by a processor a maximum bit precision for a variable for effecting a resource optimization.

7. In an electronic design automation system comprising a compiler for processing a file including a real variable, an error-analysis method comprising the step of:

determining a minimum number of bits for representing a real variable.

8. Compact electronic storage technique comprising the steps of:

identifying a plurality of arrays having a bit precision that is less than an available memory width; and

packing more than one array element into a memory location associated with the available memory.

9. In an electronic design automation system, a levelization method comprising the step of:

transforming a first statement in an intermediate-format description into a second statement associated with only one operation.

10. In an electronic design automation system, a memory-access method comprising the step of:

determining a memory access pattern or a memory access constraint for a design synthesis.

11. In an electronic design automation system, a transformation method comprising the step of:

transforming an intermediate format description for synthesizing an optimized memory access.

12. In an electronic design automation system, a method comprising the step of:

determining a number of states in a finite state machine realization associated with a transformed intermediate format description.

13. In an electronic design automation system, a method comprising the step of:

determining a finite state machine realization of one or more levelized statement in an intermediate format description.

14. In an electronic design automation system, a method comprising the step of:

determining a high-level variable or signal assignment for one or more variable in an intermediate format description.

15. In an electronic design automation system, a method comprising the step of:

identifying a pipeline opportunity in an intermediate format description.

16. In an electronic design automation system, a method comprising the step of:

transforming an intermediate format description to enable pipelining in a produced high-level code.

17. In an electronic design automation system, a method comprising the step of:

producing a pipelined high-level code from a transformed intermediate format description.